University of Zurich[UZH]

Zurich Open Repository and Archive
University of Zurich
University Library
Strickhofstrasse 39
CH-8057 Zurich
www.zora.uzh.ch

Year: 2007

A new triangular multiple flow direction algorithm for computing upslope areas from gridded digital elevation models

Seibert, Jan ; McGlynn, Brian L

Abstract: Gridded digital elevation data, often referred to as DEMs, are one of the most widely available forms of environmental data. Topographic analysis of DEMs can take many forms, but in hydrologic and geomorphologic applications it is typically used as a surrogate for the spatial variation of hydrological conditions (topographic indices) and flow routing. Here we report on a new flow routing algorithm and compare it to three common classes of algorithms currently in widespread use. The advantage of the new algorithm is that unrealistic dispersion on planar or concave hillslopes is avoided, whereas multiple flow directions are allowed on convex hillslopes. We suggest that this new triangular multiple flow direction algorithm (MD1) is more appropriate for a range of flow routing and topographic index applications.

DOI: https://doi.org/10.1029/2006WR005128

Posted at the Zurich Open Repository and Archive, University of Zurich
ZORA URL: https://doi.org/10.5167/uzh-72475
Journal Article
Published Version

Originally published at:
Seibert, Jan; McGlynn, Brian L (2007). A new triangular multiple flow direction algorithm for computing upslope areas from gridded digital elevation models. Water Resources Research, 43(4):online.
DOI: https://doi.org/10.1029/2006WR005128

A new triangular multiple flow direction algorithm for computing upslope areas from gridded digital elevation models

Jan Seibert[1] and Brian L. McGlynn[2]

Received 25 April 2006; revised 8 November 2006; accepted 22 November 2006; published 5 April 2007.

[1] Gridded digital elevation data, often referred to as DEMs, are one of the most widely available forms of environmental data. Topographic analysis of DEMs can take many forms, but in hydrologic and geomorphologic applications it is typically used as a surrogate for the spatial variation of hydrological conditions (topographic indices) and flow routing. Here we report on a new flow routing algorithm and compare it to three common classes of algorithms currently in widespread use. The advantage of the new algorithm is that unrealistic dispersion on planar or concave hillslopes is avoided, whereas multiple flow directions are allowed on convex hillslopes. We suggest that this new triangular multiple flow direction algorithm (MD∞) is more appropriate for a range of flow routing and topographic index applications.

Citation: Seibert, J., and B. L. McGlynn (2007), A new triangular multiple flow direction algorithm for computing upslope areas from gridded digital elevation models, *Water Resour. Res.*, *43*, W04501, doi:10.1029/2006WR005128.

1. Introduction

[2] Gridded digital elevation data (DEMs) are one of the most widely available types of spatially distributed environmental data. DEMs at resolutions greater than 2 m are quickly becoming available for new regions of the globe. Availability and use of digital elevation data has outpaced method development, leading to a growing gap between application and both computational efficiency and data mining techniques. Topographic analysis can take many forms, but in hydrologic and geomorphologic applications it is typically used as a surrogate for the spatial variation of hydrological conditions (topographic indices) and flow routing.

[3] Topographic indices refer to information derived from original DEMs that support evaluation and use of topographic data. Topographic indices have been used to describe spatial soil moisture patterns [e.g., *Burt and Butcher*, 1985] and many other hydrologic variables (see review by *Moore et al.* [1991]). These indices can be grouped into locally determined indices, such as elevation, slope, aspect, and curvature, as well as indices that also consider the elevations at more distant points, such as upslope accumulated area, distance to stream, or elevation above the stream, and combinations of indices, such as the topographic wetness index.

[4] In this paper we focus on the accumulated upslope area (also called local contributing area) as one important topographic index. We suggest a new algorithm for flow routing and upslope area accumulation and compare it to different existing algorithms. The underlying assumption of these flow algorithms is that the direction of downslope flow follows the surface topography and that accumulated area is a proxy for water flow. We thus consider the terms "routing of flow" and "routing of area" as synonyms.

[5] Besides other applications the upslope area is used to compute the widely used topographic wetness index (TWI) [*Beven and Kirkby*, 1979], which is a combination of the upslope area per unit contour length a, which indicates the amount of area flowing to a specific location, and the local slope $\tan\beta$, which is a measure of the potential drainage from a place, $\mathrm{TWI} = \ln(a/\tan\beta)$. The TWI can be calculated from gridded elevation data using various algorithms, which differ mainly in the way upslope area is computed [*Quinn et al.*, 1995; *Wolock and McCabe*, 1995; *Tarboton*, 1997; *Erskine et al.*, 2006]. An appropriate estimation of the upslope area is critical for the correct calculation of TWI, and we propose an improved method for this area estimation.

2. Background: Three Common Existing Algorithms

[6] In the single-direction flow algorithm, D8 [e.g., *O'Callaghan and Mark*, 1984], all the area from one cell is routed into the steepest of its eight neighboring cells. Since there is no possibility for flow to be distributed to two cells, flow tends to become concentrated to distinct, often artificially straight lines [e.g., *Erskine et al.*, 2006]. An additional problem is that the steepest gradient actually might fall between two of the eight cardinal and diagonal directions.

[7] The multiple flow direction algorithm, MD8, distributes the flow to all neighboring downslope cells weighted according to slope [*Quinn et al.*, 1991, 1995] and tends to produce more realistic looking spatial patterns than the D8 algorithm by avoiding concentration to distinct lines. The

[1]Department of Physical Geography and Quaternary Geology, Stockholm University, Stockholm, Sweden.
[2]Department of Land Resources and Environmental Sciences, Montana State University, Bozeman, Montana, USA.

Copyright 2007 by the American Geophysical Union.
0043-1397/07/2006WR005128

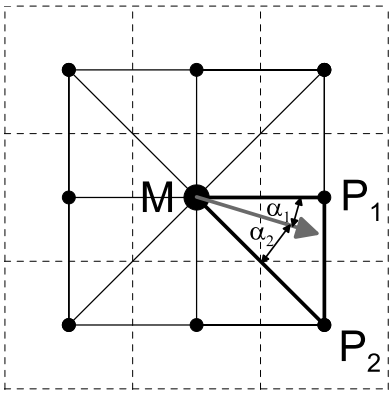


Figure 1. Triangular multiflow algorithm used to compute upslope area. Around the midpoint (M) of the pixel in question, eight planar triangular facets are constructed with midpoints (P1 and P2) of two adjacent pixels. The slope direction of each of these triangular facets is then calculated. For directions pointing between P1 and P2 the flow is distributed to the two cells on the basis of the direction of the steepest slope (i.e., the portions of flow to the two cells with P1 and P2 are $\alpha_1/45°$ and $\alpha_2/45°$, respectively).

MD8 algorithm is also more robust than D8. Using D8, a tiny elevation difference between two of the neighboring cells can have a large effect as one of the cells receives all the area. Using MD8, such elevation differences have a less influential effect because both cells receive about the same portion of the accumulated area. In other words, what might be a question of a few percent when using MD8 can be a question of 0 or 100% when using D8. The disadvantage of the multiple flow direction algorithm proposed by *Quinn et al.* [1991] is that the area from one cell is routed to all downslope cells and thus is dispersed to a large degree even for convergent hillslopes. To reduce this dispersion, *Holmgren* [1994] suggested partitioning the flow according to slope to an exponent and suggested values of 4–6 for the exponent. Even with this modification, flow pathways based on the MD8 algorithm cross each other to a large degree. This issue is less consequential (because of cross compensation) when one evaluates only the amount of area routed downslope. This crossing is more problematic when one wants to use the MD8 routing algorithm to compute the flow of substances such as solutes or sediment.

[8] *Tarboton* [1997] suggested using triangular facets to remove the limitation of only eight possible directions in single flow direction algorithms. Tarboton termed this approach D∞ to describe infinite possible single-direction flow pathways. *Tarboton*'s [1997] D∞ approach allows only a single flow direction but allows area to flow into one or two downslope cells depending on direction. *Orlandini et al.* [2003] suggested a path-based method where all accumulated area is routed toward one of the eight cardinal and diagonal directions but cumulative deviations from the actual steepest directions along the upslope path are taken into account. Other algorithms such as digital elevation model networks (DEMON) [*Costa-Cabral and Burges*, 1994] might have theoretical advantages but are too complex and case specific to be implemented for most applications [*Tarboton*, 1997].

3. New Algorithm

[9] Our new triangular multiple flow direction algorithm (MD∞), described in this paper, combines the advantages of the multiple flow direction algorithm as proposed by *Quinn et al.* [1991] with the use of triangular facets as in the D∞ approach described by *Tarboton* [1997]. Essentially, our new algorithm is a D∞ algorithm allowing for multiple-direction flow directions. In the following, we refer to the new algorithm as MD∞.

[10] For each grid cell the portion of the accumulated area that is distributed to each of its neighboring cells is computed by first determining the flow directions which receive area. As in the approach proposed by *Tarboton* [1997], triangular facets are used to compute local slope directions and gradients around the cell in question. Around the midpoint (M) of the cell, eight planar triangular facets are constructed with midpoints (P1 and P2) of two adjacent neighboring cells (Figure 1). For each of these local planes the direction of the steepest gradient is computed as follows. Let h_M, h_{P1}, and h_{P2} be the elevations of M, P1, and P2, respectively. The elevation differences between M and the two neighboring points are computed as $z_1 = h_{P1} - h_M$ and $z_2 = h_{P2} - h_M$. Similarly, x_i and y_i denote the differences in x and y coordinates between M and the two neighboring

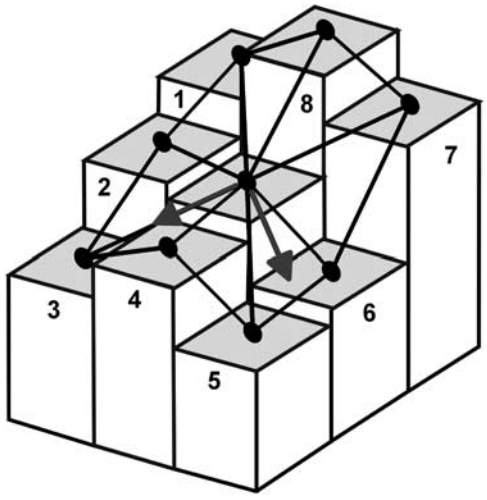
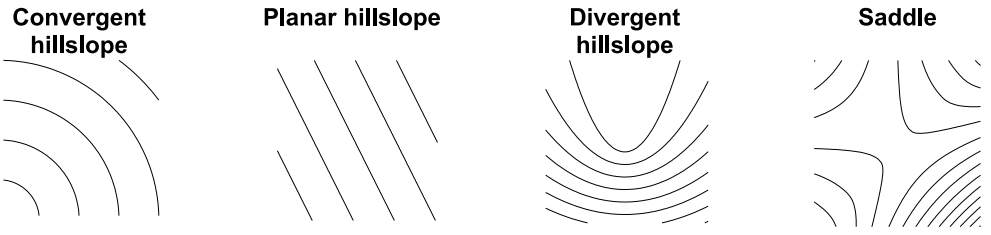


Figure 2. Example illustrating the construction of triangular facets around one cell. The direction of the steepest slope for the triangular facets may point between two neighboring cells (cells 5 and 6 in the example) or point toward a direction outside the 45° angle range of the particular triangular facet (e.g., the triangular facet defined by the midpoint and cells 4 and 5), and thus the direction is set to the steeper direction of the directions toward the two neighboring cells. In the latter case this direction only receives area if the same direction has been determined for two adjacent triangular facets. This is the case for the direction pointing toward cell 3 but not for the directions pointing toward cells 5 and 6.



Elevation

283.2	290.0	297.0	159	161	163	119.3	120.0	119.3	99.6	100.0	100.4
275.9	**282.8**	290.0	158	**160**	162	118.6	**120.0**	118.6	100.0	**100**	99.8
268.7	275.9	283.2	157	159	161	117.3	118.0	117.3	100.4	99.8	98.8

Single-direction, D8

0	0	0	0	0	0	0	0	0	0	0	0
0	**1**	0	0	**1**	0	0	**1**	0	0	**1**	0
1	0	0	1	0	0	0	1	0	0	0	1

Multiple-direction, MD8

0	0	0	0.12	0	0	0.04	0	0.04	0.19	0	0
0.29	**1**	0	0.34	**1**	0	0.12	**1**	0.12	0	**1**	0.13
0.42	0.29	0	0.37	0.17	0	0.21	0.26	0.21	0	0.13	0.55

Triangular single-direction, D∞

0	0	0	0	0	0	0	0	0	0	0	0
0	**1**	0	0.40	**1**	0	0	**1**	0	0	**1**	0
1	0	0	0.60	0	0	0.26	0.74	0	0	0	1

Triangular multiple-direction, MD∞

0	0	0	0	0	0	0	0	0	0.25	0	0
0	**1**	0	0.40	**1**	0	0	**1**	0	0	**1**	0
1	0	0	0.60	0	0	0.13	0.74	0.13	0	0	0.75

Figure 3. Distribution of accumulated area from one cell among the eight neighboring cells for four examples using synthetic data. (top) Schematic map showing the general topography. (bottom) Three-by-three panels giving the exact elevation values followed by the portions received by the neighboring cells computed using the different algorithms.

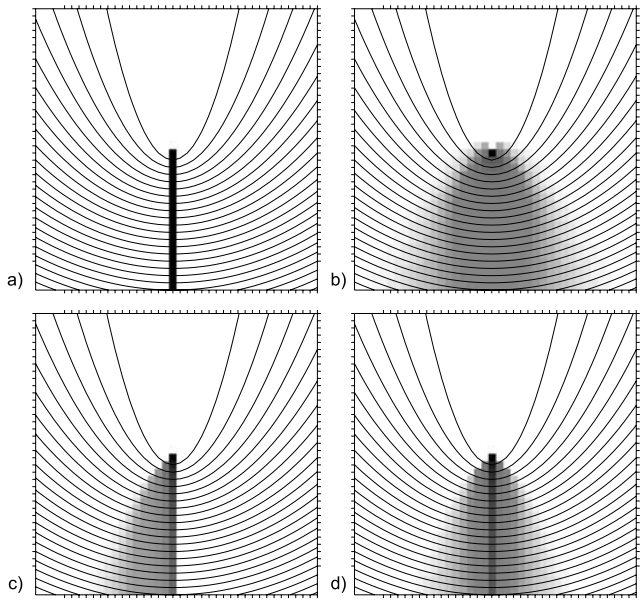


Figure 4. Downslope pattern of area distributed from one cell on a divergent hillslope (synthetic data) for the different algorithms: (a) D8, (b) MD8, (c) D∞, and (d) MD∞. The portions vary from 0 (white) to 1 (black).

points. The normal vector (i.e., the vector perpendicular to the plane) can then be computed as

$$n = \begin{bmatrix} n_x \\ n_y \\ n_z \end{bmatrix} = \begin{bmatrix} (z_1 y_2 - z_2 y_1) \\ (z_1 x_2 - z_2 x_1) \\ (y_1 x_2 - y_2 x_1) \end{bmatrix}. \quad (1)$$

The direction d and slope s of the triangular facet can then be computed using equation (2). A value of zero for d denotes the direction of the y axis, and a value of $3\pi/2$ corresponds with the direction of the x axis.

$$d = \begin{cases} 0 & n_x = 0, n_y \geq 0 \\ \pi & n_x = 0, n_y < 0 \\ \frac{\pi}{2} - \arctan\left(\frac{n_y}{n_x}\right) & n_x > 0 \\ \frac{3\pi}{2} - \arctan\left(\frac{n_y}{n_x}\right) & n_x < 0 \end{cases} \quad (2)$$

$$s = -\tan\left(\arccos\left(\frac{n_z}{\sqrt{n_x^2 + n_y^2 + n_z^2}}\right)\right). \quad (3)$$

If this steepest direction from M is outside the 45° ($\pi/4$ radian) angle range of the particular triangular facet (i.e., not between the vectors pointing from M toward P_1 and P_2, respectively), the direction with the steeper downslope gradient, of the two directions toward P_1 or P_2, is used as the steepest direction, and the slope is computed between M and P_1 or P_2. If both P_1 and P_2 have higher elevations than M, both directions, obviously, are excluded. After computing the steepest downslope directions for all eight triangular facets, those directions are maintained as the locally steepest directions that are within the 45° angle range. The directions that point toward any of the neighboring cells (i.e., toward P_1 or P_2) are only maintained when the same direction has been determined for two adjacent triangular facets. These cases are illustrated in Figure 2: The first case is the direction pointing between cells 5 and 6, and the latter case is the direction pointing toward cell 3.

[11] After the downslope directions are all computed for a cell, its accumulated area is distributed to these directions. A procedure similar to that proposed by *Quinn et al.* [1991] is used: Accumulated area is weighted and distributed downslope on the basis of the gradients. Similar to the MD8 algorithm, an exponent can be used to weight steeper directions more heavily [*Quinn et al.*, 1995]. This provides the user with the opportunity to partially control dispersion. If a downslope direction falls between two neighboring cells, the area is further distributed to these two cells according to the relative differences in direction (as suggested by *Tarboton* [1997] for the one steepest outflow) (Figure 1).

[12] The proposed MD∞ algorithm is an approach to extend the D∞ algorithm suggested by *Tarboton* [1997]. In many cases, D∞ and MD∞ provide exactly the same results (e.g., on planar or convergent hillslopes), but results differ in cases where there is more than one locally steepest downslope direction from a cell (e.g., on divergent hillslopes or along ridges).

4. Results

[13] We demonstrate the value of the new algorithm (MD∞) by comparing it with the D8, MD8, and D∞ algorithms. These comparisons are based on both synthetic elevation data and real-world DEMs. For the latter we used a low-relief example from central Sweden and a high-relief example from Montana, United States.

[14] Four examples using synthetic elevation data show some of the differences between the algorithms (Figure 3). For the convergent hillslope, all algorithms except MD8 result in the same routing where only one cell receives a portion of the area. Routing the area to only one cell in the direction of the convergent hillslope seems appropriate, whereas the distribution to three cells by MD8 seems unrealistically dispersive. Actually, two of the cells receiv-

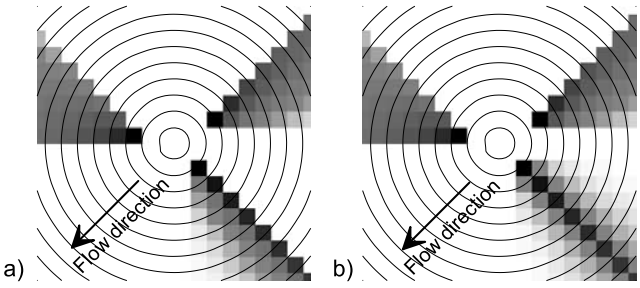


Figure 5. Downslope pattern of area distributed from cells on an outward cone (synthetic data) for two different algorithms: (a) D∞ and (b) MD∞. The portions vary from 0 (white) to 1 (black).

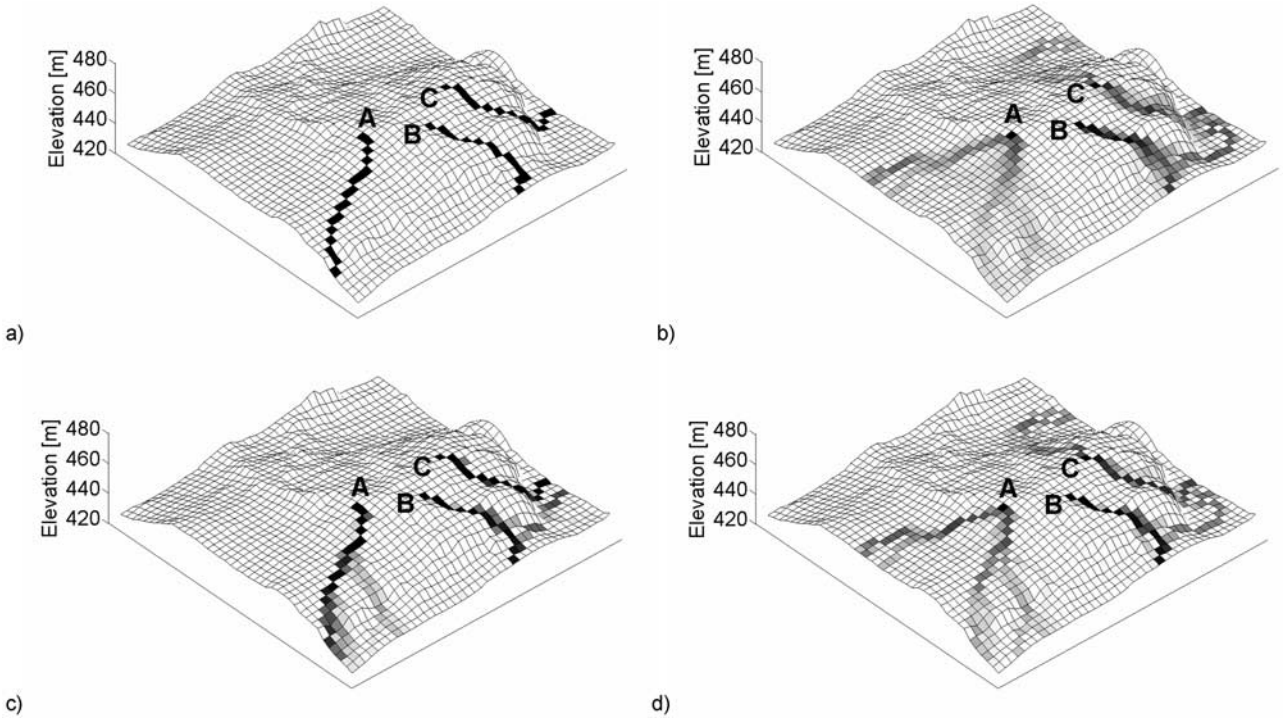


Figure 6. Downslope distribution of area accumulated in one cell using the different algorithms: (a) D8, (b) MD8, (c) D∞, and (d) MD∞. The example uses a 20 m DEM for an area in central Sweden (Middagsberget). Three cells (A, B, and C) were assigned a value of 1. The relative amount of area from any of these cells routed to downslope cells varies from 0 (white) to 1 (black).

ing area are upslope from the centerline of the hollow (although they are downslope from the middle cell). If the steepest gradient had not been aligned with one of the eight diagonal or cardinal directions, results would have differed such that both D∞ and MD∞ would have two cells receiving area.

[15] The steepest direction in the planar-hillslope example is not aligned with one of the eight diagonal or cardinal directions (Figure 3). While both D∞ and MD∞ result in one flow direction, this direction falls between two cells, and thus both cells receive area. As there is only one outflow direction, D∞ and MD∞ are exactly the same for this case. The single-direction algorithm results in a wrong direction, while the multiple-direction algorithm results in overly dispersive flow.

[16] The differences between the algorithms become most obvious for the divergent hillslope (Figure 3). D8 routes all the area toward one cell, while seven of the eight neighboring cells receive area with the MD8 algorithm. The steepest directions point somewhat to the left and right. However, since there can only be one outflow direction in the D∞ algorithm, only one of these two directions (and thus two cells) receives area, whereas both directions (and thus three cells) should receive area. These differences also result in dissimilar patterns in the distribution of the area from one cell farther downslope (Figure 4).

[17] Finally, for a cell located on a saddle both D8 and D∞ result in flow toward only the steepest direction, whereas MD8 and MD∞ allow for flow toward both downslope directions (Figure 3). More area is routed to the right since the gradient there is steeper.

[18] In the two examples above the considered cell is centered on the ridgeline (shoulder) of the divergent hillslope and on the saddle, respectively. While this might be uncommon in real-world DEMs, results would also differ between D∞ and MD∞ if the cell were not exactly centered and only a portion of the cell were located on the ridgeline or saddle.

[19] *Tarboton* [1997] compared D∞ with other algorithms using a plane, an inward cone, and an outward cone as synthetic examples. For these three examples, D∞ and

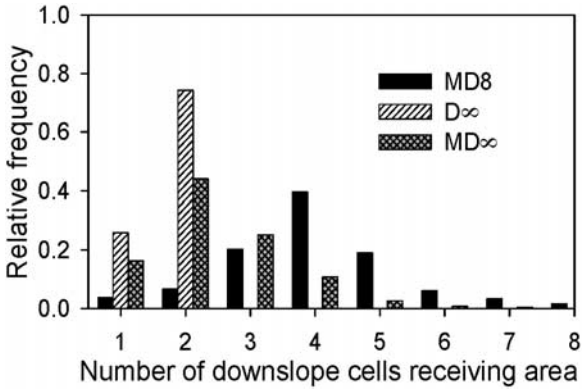


Figure 7. Distribution of the number of cells that receive accumulated area from one cell. The histograms show the distribution for all cells in an example DEM for an area in central Sweden (Middagsberget, shown in Figure 6).

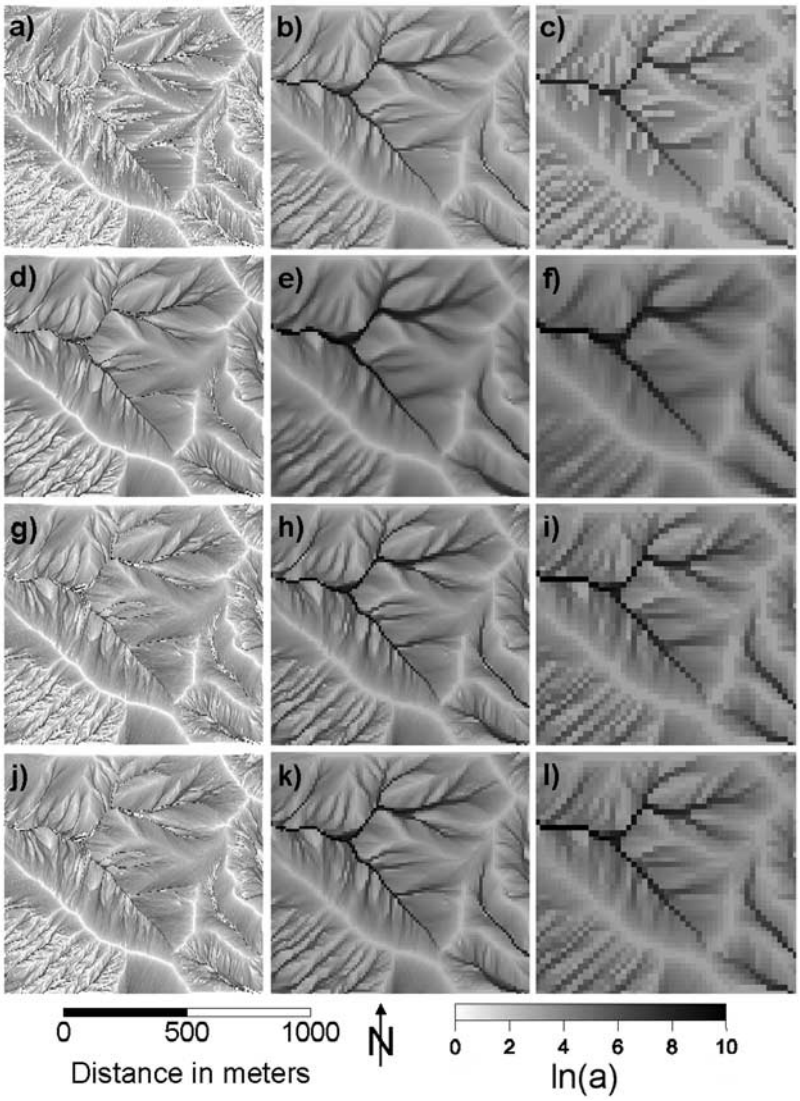


Figure 8. Maps of accumulated area per contour length a for four different flow accumulation algorithms (vertically) and three different DEM cell sizes (horizontally). The methods include (a–c) D8, (d–f) MD8, (g–i) D∞, and (j–l) MD∞. The grid cell resolutions include 2 m (Figures 8a, 8d, 8g, and 8j), 10 m (Figures 8b, 8e, 8h, and 8k), and 20 m (Figures 8c, 8f, 8i, and 8l). Corresponding histograms of the number of downslope cells receiving area are shown in Figure 9.

MD∞ provide exactly the same results with only one exception: The case of an outward cone. The steepest direction from a cell on the diagonal direction from the center of an outward cone is not exactly along the diagonal, but the slope is slightly steeper somewhat to the left and right from the diagonal direction. Only one direction is allowed in D∞, and the accumulated area is routed toward the downslope cells in the diagonal direction and one of the adjacent cardinal directions (Figure 5a). In MD∞, two directions are allowed, and both cells in the cardinal directions adjacent to the downslope diagonal direction receive some portion of the accumulated area (Figure 5b). This results in a conceptually more reasonable symmetric pattern.

[20] The differences between the different methods can also be seen when using real-world DEMs. We applied each algorithm on a 20 m DEM of a $\sim$1 km^2 region in central Sweden (Middagsberget). We computed which downslope cells receive which portion of the area accumulated in three cells located in unique topographic positions (Figure 6). D8 results in no dispersion at all (Figure 6a), while there is much dispersion with MD8 (Figure 6b). D∞ results in some dispersion, but it is only the steepest direction that receives area (Figure 6c). For the two cases in more divergent locations, MD∞ therefore gives different patterns (locations A and C in Figure 6d), whereas there is less difference in convergent locations (B).

[21] Another way to look at these differences is to compare the number of cells that receive some portion of the accumulated area. By definition, only one cell receives area for D8. With MD8 between one and eight cells can receive area; this was the case for the DEM example. On average, four cells received area, and there was a large variation (Figure 7). D∞ can have one or two cells

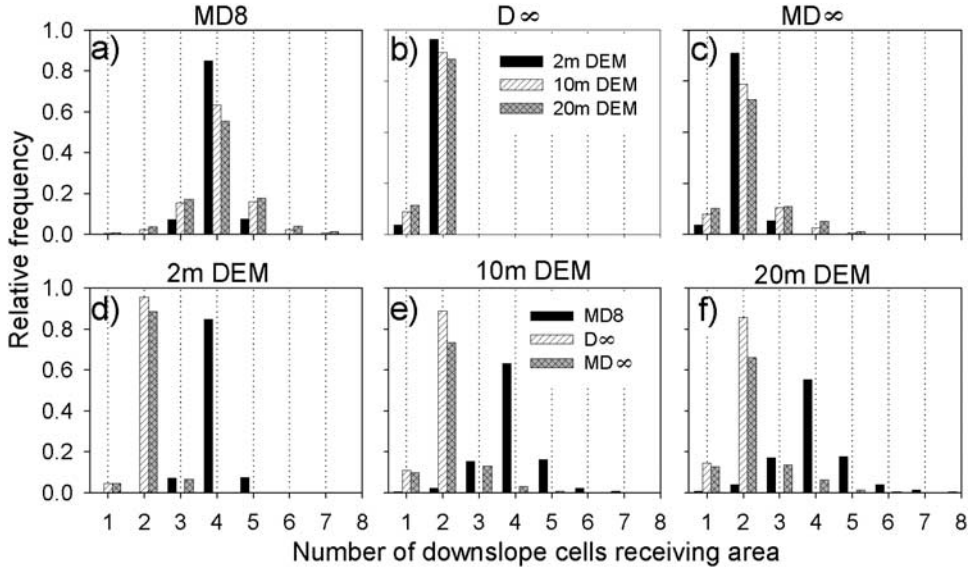


Figure 9. Histograms of the number of downslope cells receiving area shown (a–c) in terms of flow direction algorithm and (d–f) again in terms of grid cell resolution for the watershed shown in Figure 8 (Big Sky, Montana, United States).

receiving area. In our D∞ test case the area was routed to two downslope cells for the majority of cells. For MD∞, three or four receiving cells were common in addition to the most common two cells. Area was routed to more than two cells for ~40% of all cells. For ~30% of the cells, there was more than one downslope direction, and in 8% of the cases, there were three or more downslope directions. In other words, divergent hillslopes are common in real-world DEMs. This implies that allowing for more than one downslope direction (and thus for more than two downslope cells receiving area) can have a large impact on the computed flow accumulation map and any subsequent calculations or analyses.

[22] We tested each of the four algorithms (D8, MD8, D∞, and MD∞) for a high-relief section (1 km^2) of the West Fork of the Gallatin River watershed, Big Sky, Montana, United States (Figure 8). A high-resolution airborne laser swath mapping DEM (multiple returns per m^2) was available for this watershed. For the examples shown here the DEM resolution was coarsened to 2, 10, and 20 m by thinning. The West Fork area is a steeply sloping mountainous terrain. This area is a strong contrast to the gentle topography of the previous example from the low-relief region of Sweden (Figure 6).

[23] Figure 8 shows accumulated area maps for the four different flow accumulation algorithms (vertically) and three different DEM cell sizes (horizontally). The single-direction algorithm (D8) accumulated area maps (Figures 8a–8c) have linear features because of their restriction to diagonal and cardinal directions. This is most obvious in the high-resolution (2 m) DEM.

[24] The multiple-direction algorithm (Figures 8d–8f) is highly dispersive. This highly dispersive algorithm is visually appealing yet likely unrealistic. Figure 9 shows histograms of the number of downslope cells receiving area using the multiple-direction flow algorithm. Compared to the results for the low-relief DEM from Sweden, four downslope cells receiving area is even more common (~60% as compared to ~40%). For decreasing cell sizes the area is more commonly routed to four cells (Figure 9a). This is because planar 3 × 3 cell windows are more common with decreasing cell size.

[25] The D∞ algorithm limits flow to one direction, but since the angle is not restricted to the cardinal or diagonal directions, the area often is routed to two downslope cells (Figures 8g–8i). The D∞ algorithm directed flow into two downslope cells in 95, 89, and 86% of the cases with increasing cell size (Figure 9b), with the remainder of cells flowing to just one downslope cell. In other words, increasing cell size results in slightly less dispersion and more single downslope cells receiving area. The D∞ algorithm addresses the problem of the single flow direction algorithm by allowing flow in any downslope direction and addresses the overdispersion of the multiple-direction algorithm. However, D∞ is limited to only two downslope cells.

[26] The triangular multiple-direction (MD∞) algorithm resulted in flow patterns similar to the D∞ algorithm (Figures 8j–8l) but allowed downslope flow into up to eight downslope cells where appropriate, such as on convex slopes (Figures 3 and 4). The MD∞ algorithm resulted in flow into two downslope cells in 89, 74, and 66% of the cases with increasing cell size. With each algorithm, dispersion or flow into a greater number downslope cells increased with increasing cell size, except where restricted as in the single-direction (one cell) and D∞ (two cells) algorithms.

[27] Figure 10 shows the spatial pattern of differences in accumulated area computed using the D∞ and MD∞ algorithms. The difference map (Figure 10a) demonstrates the impact of only one flow direction in the D∞ approach. For example, if a cell is located on a subwatershed divide (ridge or shoulder slope), the D∞ algorithm will allow area to flow in only one direction, while the MD∞ algorithm will allow area to flow in multiple downslope directions. As a result, convergent areas in Figure 10a are either red or blue, depending on which downslope direction was chosen

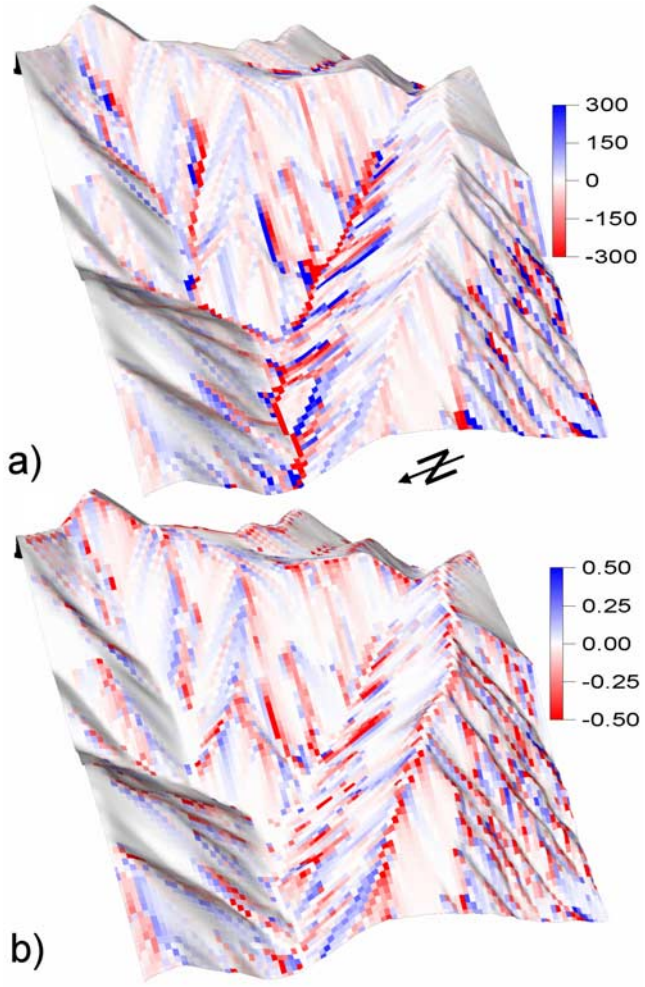


Figure 10. Map of (a) absolute (m^2) and (b) relative (dimensionless) accumulated area differences between the D∞ algorithm and the MD∞ algorithm. Figure 10a is the D∞ accumulated area minus the MD∞ accumulated area. Figure 10b is Figure 10a divided by the D∞ accumulated area.

in upslope areas by the D∞ algorithm. The impact of this in terms of area is most apparent where the greatest area has been accumulated, such as lower regions of the watershed. This is clear in Figure 10b, where the difference in accumulated area has been normalized by the accumulated area calculated from the D∞ approach. In this map it is apparent on nearly all convex slopes that one side of each divide is blue and the other side of the divide is red. This occurs not only along ridges at the subwatershed divides but also along shoulders along lower hillslopes. This phenomenon is also seen for synthetic data in Figure 4 and for a low-relief case in Figure 6 and is simply a result of D∞ allowing flow in only one downslope direction.

5. Concluding Remarks

[28] We analyzed four algorithms with synthetic data (Figures 3–5), with DEM data from low-relief (Figures 6 and 7) and high-relief regions (Figures 8–10), and across a range of cell sizes, including 2, 10, and 20 m^2. The limitations of single flow direction algorithms that allow flow into only one downslope cell are well acknowledged. The limitation of flow via cardinal and diagonal directions alone and excess flow dispersion were addressed by *Tarboton* [1997] and the D∞ algorithm. However, D∞ is still limited to a single flow direction and two downslope cells. This becomes increasingly important on convex slopes, as demonstrated by both synthetic and real-world DEM data. On these slopes the D∞ suffers from too little flow dispersion when there are different flow directions from one cell, as illustrated in Figures 4, 6, and 10. The MD8 algorithm, on the other hand, suffers from high flow dispersion even on convergent hillslopes. The triangular multiple-direction algorithm (MD∞) we present here is an evolution of existing algorithms that allows multidirectional flow in any downslope direction, thereby combining the benefits of D∞ and the multiple-direction algorithms. We suggest that the MD∞ algorithm is more appropriate than the existing flow algorithms across a range of landscapes, DEM resolutions, and applications.

[29] **Acknowledgments.** This work was made possible by the National Science Foundations Hydrologic Sciences Program (EAR-0337650), the Integrated Carbon Cycle Research Program (EAR-0404130), and the Geography and Regional Science Program (BCS-0518429) grants awarded to B.L.M. J.S. received support from the Swedish Research Council. Topographic data for the example from Sweden were provided by Ursula Zinko.

References

Beven, K. J., and M. J. Kirkby (1979), A physically based, variable contributing area model of basin hydrology, *Hydrol. Sci. Bull.*, *24*, 43–69.

Burt, T. P., and D. P. Butcher (1985), Topographic controls of soil moisture distributions, *Eur. J. Soil Sci.*, *36*(3), 469–486.

Costa-Cabral, M., and S. J. Burges (1994), Digital elevation model networks (DEMON): A model of flow over hillslopes for computation of contributing and dispersal areas, *Water Resour. Res.*, *30*(6), 1681–1692.

Erskine, R. H., T. R. Green, J. A. Ramirez, and L. H. MacDonald (2006), Comparison of grid-based algorithms for computing upslope contributing area, *Water Resour. Res.*, *42*, W09416, doi:10.1029/2005WR004648.

Holmgren, P. (1994), Multiple-flow direction algorithms for runoff modelling in grid based elevation models: An empirical evaluation, *Hydrol. Processes*, *8*, 327–334.

Moore, I. D., R. B. Grayson, and A. R. Ladson (1991), Digital terrain modeling: A review of hydrological, geomorphological, and biological applications, *Hydrol. Processes*, *5*, 3–30.

O'Callaghan, J. F., and D. M. Mark (1984), The extraction of drainage networks from digital elevation data, *Comput. Vision Graphics Image Process.*, *28*, 328–344.

Orlandini, S., G. Moretti, M. Franchini, B. Aldighieri, and B. Testa (2003), Path-based methods for the determination of nondispersive drainage directions in grid-based digital elevation models, *Water Resour. Res.*, *39*(6), 1144, doi:10.1029/2002WR001639.

Quinn, P. F., K. J. Beven, P. Chevallier, and O. Planchon (1991), The prediction of hillslope flowpaths for distributed modelling using digital terrain models, *Hydrol. Processes*, *5*, 59–80.

Quinn, P. F., K. J. Beven, and R. Lamb (1995), The ln (a/tanβ) index: How to calculate it and how to use it within the TOPMODEL framework, *Hydrol. Processes*, *9*, 161–182.

Tarboton, D. G. (1997), A new method for the determination of flow directions and upslope areas in grid digital elevation models, *Water Resour. Res.*, *33*(2), 309–319.

Wolock, D. M., and G. J. McCabe (1995), Comparison of single and multiple-flow direction algorithms for computing topographic parameters in TOPMODEL, *Water Resour. Res.*, *31*(5), 1315–1324.

B. L. McGlynn, Department of Land Resources and Environmental Sciences, Montana State University, Bozeman, MT 59717, USA.

J. Seibert, Department of Physical Geography and Quaternary Geology, Stockholm University, Svante Arrhenius väg 8C, SE-10691 Stockholm, Sweden. (jan.seibert@natgeo.su.se)